ELSEVIER
Available online at www.sciencedirect.com
ScienceDirect
International Journal of Approximate Reasoning
48 (2008) 23–35
INTERNATIONAL JOURNAL OF APPROXIMATE REASONING
www.elsevier.com/locate/ijar

Refinement of temporal constraints in fuzzy associations

Thomas Sudkamp *

Department of Computer Science, Wright State University, Dayton, OH 45435, USA

Accepted 14 June 2007
Available online 1 July 2007

Abstract

The objectives of hypothesis refinement in knowledge discovery are to produce rules that more accurately model the underlying data while maintaining rule interpretability. In this paper we introduce two refinement strategies for association rules with fuzzy temporal constraints. Disjunctive generalization produces more general rules by merging adjacent constraints within a partition of the window of temporal relevance. Temporal specification uses linguistic hedges to reduce the duration of a constraint to better model the distribution of examples. Both types of refinement produce rules expressible using the linguistic terms of the original rules. The acquisition of the information needed to perform the refinements is incorporated into a general algorithm for determining the number of examples and counterexamples of rules with fuzzy temporal constraints.

Keywords: Temporal association rules; Data mining; Temporal relevance; Rule refinement, Fuzzy sets

1. Introduction

When machine learning and data mining are employed to assist human decision making, interpretability is a critical feature of the relationships produced by the discovery process. For relationships expressed as rules, interpretability requires that the predicates in a rule be given in linguistic terms that utilize standard or agreed upon meanings. Because of the ability to model linguistic terms, the use of fuzzy representations and reasoning methodologies in temporal data mining has recently been identified as one of the challenging but promising machine learning research topics [1].

Constructing rules from a fixed set of linguistic terms produces a conflict between accuracy and interpretability in the discovery process. Learning techniques based solely on clustering or statistical analysis of the distribution of the data cannot be expected to yield relationships that are readily expressible in the language of the problem domain. These approaches, however, have the advantage of producing results completely determined by the data itself without being influenced by a bias imposed by language restrictions. A challenge in producing interpretable rules is modelling the data as well as possible within the constraints of a predefined

* Tel.: +937 775 5118; fax: +937 775 5133.
E-mail address: thomas.sudkamp@wright.edu

0888-613X/$ - see front matter
doi:10.1016/j.ijar.2007.06.004

language. This paper introduces two refinement strategies to produce rules that better reflect the underlying data while using the terms of the original rules to maintain linguistic expressibility.

The primary use of fuzzy sets in knowledge discovery has focused on providing linguistic representations of predicates and avoiding unnatural boundaries between classes in the discovery of quantitative association rules [2–5]. The most common applications of data mining with temporal information attempt to discover sequential patterns [6–8] or cyclic association rules [9–11] within the data. In mining sequential patterns, a time interval is used to define a sliding window that acts as a constraint upon the relevance of events. In cyclic associations, the antecedent of a rule limits the consideration of events to specific time periods. For example, an analysis of restaurant orders may produce the relationship "between 10:00pm and 1:00am, beer and pizza are ordered together." Fuzzy sets have been used to provide gradual transitions to the time intervals used in both learning sequential patterns [12] and cyclic association rules [13].

We will use fuzzy predicates to represent properties of events, and temporal relevancy will be represented by fuzzy constraints. Algorithms for discovering rules with fuzzy temporal constraints were presented in [14,15] and applied to the analysis of the concentration of oxygen in the bloodstream after use of a respirator [16].

Typical knowledge discovery algorithms perform three tasks: granularization, summarization, and hypothesis analysis. Granularization and summarization reduce the quantity of data that must be examined during the analysis of the hypotheses. After the assessment of hypotheses, refinement may be performed to improve the accuracy or the interpretability of a rule. In this paper we introduce two rule refinement strategies, disjunctive generalization and temporal specification. Disjunctive generalization produces more general rules by merging temporal constraints. Temporal specification uses linguistic hedges to reduce the duration of a constraint to better match the distribution of the data. When refinement is included in rule discovery, the summarization step must generate and retain sufficient information to efficiently assess the validity of the rules created in the refinement phase.

This paper begins by describing a representation for rules with fuzzy temporal constraints and an algorithm to identify examples and counterexamples of these rules in a temporal event stream. We then introduce the two types of rule refinement, disjunctive generalization and temporal specification, and identify the information needed to assess the rules generated by the refinement process. The acquisition of this data is incorporated into the original search procedure to efficiently assess the validity of the rules created by the refinement. The paper concludes by demonstrating the impact of refinement on rule expressibility.

2. Constraint and rule representations

In this paper a temporal pattern is represented as a rule with relevancy constraints that limit the relevance of the occurrence of one event to that of another. Two types of relevancy constraints may occur in a rule: antecedent constraints and implicative constraints. An implicative constraint links the satisfaction of the antecedent of a rule with that of the consequent. The relationship described by the rule "if Jacques arrives, Tony will leave within three days" has a window of relevance of three days; a sequence of events satisfies this rule only if Tony's departure is within three days of Jacques' arrival. An antecedent constraint specifies a time interval within which all the predicates in the antecedent of a rule must be satisfied. The condition in the rule "if British Air Flight 184 and United Flight 930 arrive within of 20 min of each other, there will be a long line at customs" provides an example of an antecedent constraint. The antecedent is satisfied only if the two flights arrive within the relevancy window of 20 min.

The role of the fuzzy predicates and constraints in temporal reasoning is illustrated by a linguistic description of traffic flow. The rule corresponding to the statement "simultaneous heavy traffic volume on route 70 and medium traffic volume on route 75 result in long delays at the exit 23 within an hour" has fuzzy predicates 'heavy volume' and 'medium volume' in the antecedent to describe traffic flow. The consequent is the fuzzy set 'long delay' describing the delay time. The implicative constraint 'within an hour' links the occurrence of antecedent conditions with the resulting congestion. The antecedent constraint, indicated by the term simultaneous, could be interpreted literally to mean at the exact moment or, more realistically, by a fuzzy set that represents 'at approximately the same time.'

The increased granularity obtained by partitioning a window of relevance into subintervals produces rules that have the ability to more accurately reflect the distribution of events within the window. For example, an implicative constraint with a window of relevance of three days may be partitioned by the temporal predicates whose linguistic interpretations are ‘within the first day after’, ‘in the second day after’, and ‘in the third day after’. Fig. 1a depicts a crisp partition of the three-day interval, where T_f represents ‘in the fth day after’. A fuzzy partition that covers this interval is given in Fig. 1b. The gradual transition between fuzzy sets avoids the boundary bias problem that may occur when a crisp partition is used for rule discovery [17]. Note that the duration of the fuzzy partition extends beyond 72 h due to the gradual transition from the fuzzy set that represents ‘in the third day after’. We will use ‘occurring-after’ conditions to represent implicative constraints and let $T_1, \ldots, T_p$ denote the partition of the window of implicative relevance.

An antecedent constraint begins at the time of the satisfaction of one of the predicates in the antecedent. Like an implicative constraint, an antecedent constraint can be decomposed into a set of crisp or fuzzy subintervals. Fig. 2 shows a decomposition of an antecedent constraint of ‘within 20 min’ into a family of nested fuzzy sets. The fuzzy set W_1 represents a fuzzy constraint of ‘within 5 min’ given by

$$W_1(x) = \begin{cases} 1 & \text{if } x \leqslant 5, \\ -0.5x + 3.5 & \text{if } 5 < x \leqslant 7, \\ 0 & \text{otherwise.} \end{cases}$$

The dotted line labelled W_g indicates the support of the fuzzy constraint W_g. The inclusion relationships $W_1 \subseteq W_2 \subseteq W_3$ are evident in Fig. 2. A family of antecedent constraints will be denoted as $W_1 \subseteq W_2 \cdots \subseteq W_q$.

Rule interpretability imposes a limit on the number of predicates that can occur in the antecedent of a rule. For the purpose of illustrating the relationships between data summarization, rule discovery, and rule refinement, we will consider the generation of rules with three predicates in the antecedent. The predicates describe the properties of n attributes $\mathscr{A}_1, \ldots, \mathscr{A}_n$ and are denoted as

Attribute	Associated Predicates
$\mathscr{A}_1$:	$A_{1,1}, \ldots, A_{1,t1}$
$\mathscr{A}_2$:	$A_{2,1}, \ldots, A_{2,t2}$
$\vdots$	$\vdots$
$\mathscr{A}_n$:	$: A_{n,1}, \ldots, A_{n,tn}.$

In the traffic flow example given above, $\mathscr{A}_i$ may represent the traffic conditions of a particular location with $A_{i,1} = low$, $A_{i,2} = medium$, and $A_{i,3} = high$ being the fuzzy predicates that describe traffic volume at location $\mathscr{A}_i$. The consequent of the rules will be a predicate $C_1, \ldots, C_m$ that describes the state of an output domain $\mathscr{C}$.

Using the notation for the domain partitions and constraints defined above, rules are written as

$$C_k : T_f\text{-after } \langle \{A_{i1,r1}, A_{i2,r2}, A_{i3,r3}\}, W_g \rangle,$$

where each of the predicates in the antecedent $\{A_{i1,r1}, A_{i2,r2}, A_{i3,r3}\}$ describes a different attribute. The interpretation of this rule is that if events satisfying $A_{i1,r1}$, $A_{i2,r2}$, $A_{i3,r3}$ occur within time W_g of each other, then C_k will occur T_f time later.

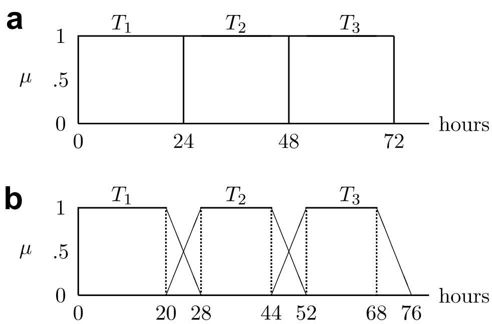

Fig. 1. Crisp and fuzzy implicative constraints.

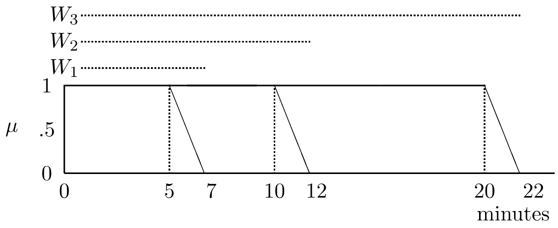

Fig. 2. Fuzzy antecedent constraints.

There at least $C(n,3) \cdot t^3 \cdot m \cdot p \cdot q$ rules of the preceding form, where t is the minimum number of predicates in an antecedent domain partition. The number of rules to be considered, and hence the work required in the discovery process increases with the number of fuzzy sets in the temporal partitions and the number of predicates.

3. Examples, counterexamples, and assessing rule validity

The classical approach for measuring rule validity is based on the number of examples and counterexamples of the rules [18,19]. The support for a rule 'if A then B' is determined by the number of items in the database in which A and B are both true. A rule whose support exceeds a predefined support threshold is said to be *frequent*. The confidence of 'if A then B' is the conditional probability of both A and B being satisfied given that A is satisfied. A frequent rule whose confidence exceeds a confidence threshold is accepted. The task of association rule mining is to efficiently determine the support and confidence values for possible rules.

Learning temporal rules requires determining the number of examples and counterexamples occurring in the event stream. The event stream consists of a sequence of events, each of which has an associated time of occurrence. Events will be denoted as $e_1, e_2, \ldots, e_i, \ldots$, and the time of the occurrence of event e_i is denoted as $t(e_i)$.

Intuitively, an example of a rule 'C_k: T_f-after $\langle\{A_{i1,r1}, A_{i2,r2}, A_{i3,r3}\}, W_g\rangle$' consists of the occurrence of events e_i, e_j, e_k in any order within the W_g antecedent time constraint that satisfy predicates A_{i_1,r_1}, A_{i_2,r_2}, and A_{i_3,r_3}, followed by the occurrence of an event c that satisfies C_k within the implicative time constraint T_f. In a similar manner, the occurrence of events e_i, e_j, e_k within the antecedent constraint that satisfy predicates A_{i_1,r_1}, A_{i_2,r_2}, and A_{i_3,r_3} that is not followed by an event that satisfies C_k is a counterexample to rule. Three events $\{e_i, e_j, e_k\}$ that satisfy the predicates in the antecedent of a rule will be referred to as an *event set* for that rule.

With fuzzy predicates and constraints, a sequence of events may be a partial example or counterexample of a rule. The degree to which a set of events constitutes an example of a rule 'C_k: T_f-after $\langle\{A_{x,r_1}, A_{y,r_2}, A_{z,r_3}\}, W_g\rangle$' is determined by the degree of satisfaction of each of the predicates and constraints in the rule. That is, the degree to which events e_i, e_j, e_k, e_c are an example of the preceding rule is $A_{x,r_1}(e_i) \otimes A_{y,r_2}(e_j) \otimes A_{z,r_3}(e_k) \otimes C_k(e_c) \otimes T_f(d_1) \otimes W_g(d_2)$, where d_1 is the time duration between the completion of the events in the antecedent and the occurrence of e_c, d_2 is the time separating the antecedent events, and $\otimes$ is a T-norm. The same events provide a counterexample to the rule to degree $A_{x,r_1}(e_i) \otimes A_{y,r_2}(e_j) \otimes A_{z,r_3}(e_k) \otimes (1 - C_k(e_c)) \otimes T_f(d_1) \otimes W_g(d_2)$. The product T-norm is normally used to determine the degree to which events constitute an example of a fuzzy rule [20].

When an example consists of the satisfaction of several temporally related events, it is possible that multiple event sets within the window of relevance satisfy the antecedent of a rule. In this case, it is necessary to specify conditions that determine which set of events constitutes an example of the rule. Consider a set of rules with an antecedent constraint of 10 h and an implicative constraint of within 24 h of the satisfaction of the antecedent (T_1 in Fig. 1). These constraints define rules of the form

$$C_k : \text{in the first day after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, \text{within 10 h}\rangle.$$

The events in Fig. 3 illustrate the choices that need to be made to determine the number of examples in an event stream.

Event	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}
Satisfies	A_{i_1,r_1}	A_{i_1,r_1}	A_{i_2,r_2}	A_{i_3,r_3}	A_{i_1,r_1}	C_k	A_{i_2,r_2}	A_{i_3,r_3}	A_{i_1,r_1}	C_k
Hours	0	2	6	8	10	13	15	18	20	30
Link 1	$\uparrow$		$\uparrow$	$\uparrow$		$\leftarrow$				
Link 2							$\uparrow$	$\uparrow$	$\uparrow$	$\leftarrow$

Fig. 3. Linkage criteria.

The event sets $\{e_1, e_3, e_4\}$, $\{e_2, e_3, e_4\}$, and $\{e_3, e_4, e_5\}$ all satisfy the antecedent constraint. Moreover, an event e_6 satisfies the consequent C_k within the time limitation set by the implicative constraint. Consequently, determining examples solely by the satisfaction of constraints would produce three examples from the single consequent e_6. This approach may produce more examples of a rule than there are instances of the consequent in the event stream. To ensure that this does not occur, we limit a consequent event to contributing to at most one example for each rule.

When there are multiple events within a window that could be linked, we will determine links as follows:

(i) Each consequent event e_c can be linked to an event set that satisfies an antecedent $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$ at most once. It may, however, be linked to many distinct antecedents.
(ii) If there are multiple instances of event sets that satisfy an antecedent and occur within the window of relevance of a consequent event c, the consequent event is linked to the earliest satisfying event set.
(iii) If a consequent event e_c that satisfies C_k is linked to an event set $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$, all elements in a subsequent linking of an event set that satisfies $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$ to an event that satisfies C_k must occur after $t(e_c)$.

Following these conventions, the consequent event at e_6 in Fig. 3 is linked to event set $\{e_1, e_3, e_4\}$. Event e_{10} is linked to the event set $\{e_7, e_8, e_9\}$, all of whose members occurred after $t(e_6)$. The decisions on linking are unique to particular problem and linking protocols must be established based on the problem domain under consideration.

We now present an algorithm for sequentially identifying all examples of rules subject to the linking conditions. Processing the event stream is accomplished using four pointers: p_1, p_2, and p_3 point to three events that make up an event set and p_c points an event that satisfies the consequent.

(1) Set $i = 1$.
(2) Set p_1 to event e_i. Let $\mathscr{A}_x$ be the domain of e_i.
(3) Set p_2 to e_{i+1} and sequentially examine the event stream with p_2 until either
  (a) an event e_j from domain $\mathscr{A}_y$, $y \neq x$, is encountered, or
  (b) an event is encountered whose occurrence is outside of the time of e_i plus the maximum antecedent constraint time.
(4) If (b), set i to $i + 1$, and continue with step 2.
(5) Set p_2 to event e_j.
(6) Set p_3 to e_{j+1} and sequentially examine the event stream with p_3 until either
  (a) an event e_k from domain $\mathscr{A}_z$, $z \neq x$ and $z \neq y$, is encountered, or
  (b) an event is encountered whose occurrence is outside of the time of e_i plus the maximum antecedent constraint time.
(7) If (b), set i to $i + 1$, and continue with step 2.An event set $\{e_i, e_j, e_k\}$ has been identified for rules whose antecedent consists of predicates from domains $\mathscr{A}_x$, $\mathscr{A}_y$, and $\mathscr{A}_z$.

Domain	…	$\mathcal{A}_x$	…	$\mathcal{A}_y$	…	$\mathcal{A}_z$	…	$\mathcal{C}$	…
Event	…	e_i	…	e_j	…	e_k	…	e_c	…
		$\uparrow$		$\uparrow$		$\uparrow$		$\uparrow$	
Pointers		p_1		p_2		p_3		p_c	

Fig. 4. Pointer configuration.

(8) Set p_c to e_{k+1}.
(9) Sequentially examine the event stream with p_c until either
  (a) an event e_c from domain $\mathscr{C}$ is encountered, or
  (b) an event is encountered whose occurrence is outside of the time of e_k plus the maximum implicative constraint time.
(10) Update the summarization tables with the information needed to evaluate the validity of the rules.
(11) If $t(e_c)$ is less than the time of e_k plus the maximum implicative constraint time, set p_c to e_{c+1} and continue with step 9.
(12) If $t(e_k)$ is less than the time of e_i plus the maximum antecedent constraint time, set p_3 to e_{k+1} and continue with step 6.
(13) If $t(e_j)$ is less than the time of e_i plus the maximum antecedent constraint time, set p_2 to e_{j+1} and continue with step 3.
(14) Set i to $i+1$ and continue with step 2.

The examination of events continues until one of the pointers reaches the end of the event stream. Fig. 4 shows the relationship among the pointers when an event set and consequent have been identified in step 9. The repositioning of pointers in steps 11–14 ensure that all combinations of event sets and consequences that satisfy the relevancy constraints are examined.

The objective of the summarization phase of the discovery process is to reduce the amount of data to be analyzed during rule generation and assessment. The particular information to be extracted, which was left unspecified in step 10, varies based upon the method of rule assessment, the linking protocols, and the type of rule refinement employed. This omission will be remedied in the following two sections.

The limitation to a fixed number of predicates in the antecedent permits the efficient sequential processing of the event stream described above. If the event stream contains n_e events with n_A and n_C the maximum number of events that can occur within the antecedent and implicative relevancy windows, respectively, the number of combinations of events examined is $n_e \cdot n_A \cdot (n_A - 1) \cdot n_C = O(n_e^4)$.

4. Disjunctive generalization

Disjunctive generalization extends the scope of a rule by merging adjacent fuzzy sets in a fuzzy partition of a domain. Linguistic interpretability is maintained since the new rule is expressible as a disjunction of the terms in the original rules. To illustrate disjunctive generalization, we consider the generalization of implicative constraints defined by a fuzzy partition.

Three tables are needed to compute the information to assess the validity of the original rules and to perform the generalization. Tables *Ex* and *Cex* record the number of examples and counterexamples of the original rules, respectively. *Tex* records the time of the consequent of the most recent example of a rule to enforce the linking conditions. The *Ex*, *Cex*, and *Tex* tables may be implemented using $n+3$ dimensional arrays, where n is the number of attributes.

The first n indices specify the predicates in the antecedent of a rule. If the index of dimension i, $1 \leqslant i \leqslant n$, is 0, then the antecedent does not contain a predicate concerning attribute $\mathscr{A}_i$. If index i is $r_i > 0$, the antecedent contains the predicate A_{i,r_i}. The final three indices designate the predicate in the consequent of the rule, the implicative constraint, and the antecedent constraint. Thus, the information for a rule of the form

$$C_k : T_f\text{-after } \langle\{A_{x,r1}, A_{y,r2}, A_{z,r_3}\}, W_g\rangle,$$

with $x < y < z$, is stored in the position

$$\begin{array}{ll} \text{position}: 1 & \quad x \qquad\qquad y \qquad\qquad z \qquad\qquad n \\ \text{value}: & [0,\ldots,0,r_1,0,\ldots,0,r_2,0,\ldots,0,r_3,0,\ldots,0,k,f,g] \end{array}$$

in the arrays *Ex*, *Cex*, and *Tex*. We will denote this set of indices $((x, r_1),(y, r_2),(z, r_3),(k, f, g))$.

To illustrate the use of indices to represent rules, consider the rules generated from five attribute domains $\mathcal{A}_1,\ldots,\mathcal{A}_5$ each of which has three associated predicates. The arrays *Ex*, *Cex*, and *Tex* have eight dimensions. The entry

$$Ex[0, 1, 0, 2, 2, 3, 1, 2]$$

contains the number of examples of the rule

$$C_3 : T_1\text{-after } \langle\{A_{2,1}, A_{4,2}, A_{5,2}\}, W_2\rangle.$$

The 0's in index 1 and 3 indicate that the rule antecedent does not contain predicates from domains $\mathcal{A}_1$ and $\mathcal{A}_3$. The values 1, 2, and 2in indices 2, 4 and 5 indicate that the antecedent consists of the predicates $A_{2,1}$, $A_{4,2}$, and $A_{5,2}$, respectively. The final three indices designate the consequent as C_3, the implicative constraint as T_1, and the antecedent constraint as W_2.

The search algorithm in Section 3 identifies all sequences of events that satisfy the antecedent and consequent constraints. All that remains is the determination of the number of examples and counterexamples (which completes step 10). For each event set $\{e_i, e_j, e_k\}$ produced in step 6(a), we find each combination of predicates A_{x,r_1}, A_{y,r_2}, A_{z,r_3} with $A_{x,r_i}(e_i) > 0$, $A_{y,r_2}(e_j) > 0$, and $A_{z,r_3}(e_k) > 0$. That is, we find all rule antecedents that are partially satisfied by the events. The entries in *Ex*, *Cex*, and *Tex* are updated for each such rule. Three cases may occur:

Case 1. No event e_c is discovered in step 7 within the time of the antecedent constraint. In this case, the event set is a counterexample of the rule and $Cex((x, r_1),(y, r_2),(z, r_3),(k, f, g))$ is incremented by $A_{x,r_1}(e_i) \otimes A_{y,r_2}(e_j) \otimes A_{z,r_3}(e_k) \otimes W_g(d_2)$ for each k, f, and g, where d_2 is the time separating the events e_i, e_j, and e_k.

For the cases in which an event e_c is discovered within the implicative relevance window, let d_1 be the time duration between the completion of the events in the antecedent and the occurrence of e_c and let d_2 be the time separating the antecedent events. The events e_i, e_j, e_k, e_c satisfy the rule

$$C_k : T_f\text{-after } \langle\{A_{x,r_1}, A_{y,r_2}, A_{z,r_3}\}, W_g\rangle$$

to degree $\alpha = A_{x,r_1}(e_i) \otimes A_{y,r_2}(e_j) \otimes A_{z,r_3}(e_k) \otimes C_k(e_c) \otimes T_f(d_1) \otimes W_g(d_2)$. Similarly, they are a counterexample to degree $\beta = A_{x,r_1}(e_i) \otimes A_{y,r_2}(e_j) \otimes A_{z,r_3}(e_k) \otimes (1 - C_k(e_c)) \otimes T_f(d_1) \otimes W_g(d_2)$.

Case 2. The time recorded in $Tex((x, r_1),(y, r_2),(z, r_3),(k, f, g))$ is greater than the time of at least one of e_i, e_j, or e_k. In this case, the consequent has previously been linked to an antecedent set of this type. These events are not considered to be either an example or a counterexample in accordance with the linking conditions and no action is taken.

Case 3. The time recorded in $Tex((x, r_1), (y, r_2), (z, r_3), (k, f, g))$ is less than the time of e_i, e_j, and e_k. In this case, α is added to $Ex((x, r_1),(y, r_2),(z, r_3),(k, f, g))$, β is added to $Cex((x, r_1), (y, r_2), (z, r_3), (k, f, g))$, and $Tex((x, r_1), (y, r_2), (z, r_3), (k, f, g))$ is set to the time of e_c.

After the event stream is processed, the *Ex* and *Cex* tables can be used to assess the validity of all potential rules. The number of examples of the rule

$$C_k : T_f\text{-after } \langle\{A_{x,r_1}, A_{y,r_2}, A_{z,r_3}\}, W_g\rangle$$

is $Ex((x, r_1), (y, r_2), (z, r_3), (k, f, g))$. The confidence is given by

$$\frac{Ex((x,r_1),(y,r_2),(z,r_3),(k,f,g))}{Ex((x,r_1),(y,r_2),(z,r_3),(k,f,g)) + Cex((x,r_1),(y,r_2),(z,r_3),(k,f,g))}.$$

Disjunctive generalization uses the natural hierarchy associated with fuzzy partitions to merge the relevancy constraints of rules that are accepted during the original learning process [14]. The extended constraint constructed from T_f and T_{f+1} is obtained using the Lukasiewicz T-conorm

$$T_f \cup T_{f+1}(d) = \max\{1, T_f(d) + T_{f+1}(d)\}.$$

The fuzzy set that results from merging adjacent constraints is illustrated in Fig. 5. The linguistic interpretation of the generalization is the constraint 'T_1 or T_2 after'. The requirement that both of the rules in a merger are frequent ensures that the examples supporting both of the constituent rules contribute significantly to the set of examples of the extended rule.

The support and confidence of the rule

$$C_k : T_f \text{ or } T_{f+1}\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$$

obtained by disjunctive generalization of the rules

$$C_k : T_f\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$$

and

$$C_k : T_{f+1}\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$$

can be obtained directly from the Ex and Cex tables. The number of examples is

$$\sum_{i=f}^{f+1} Ex((x,r_1),(y,r_2),(z,r_3),(k,i,g))$$

and the confidence is

$$\frac{\sum_{i=f}^{f+1}(Ex((x,r_1),(y,r_2),(z,r_3),(k,i,g)))}{\sum_{i=f}^{f+1}(Ex((x,r_1),(y,r_2),(z,r_3),(k,f,g))) + Cex((x,r_1),(y,r_2),(z,r_3),(k,f,g)))}.$$

The process of merging proceeds using the newly constructed constraints and rules. Fig. 5b shows the result of merging the three adjacent constraints given in Fig. 1.

Disjunctive generalization increases the number of rules that can be generated and analyzed. When the partition consists of n fuzzy sets, there are $(n-1)!$ possible disjunctive generalizations and the resulting increase in the expressiveness of the rules requires no additional work in the determination of examples and counterexamples.

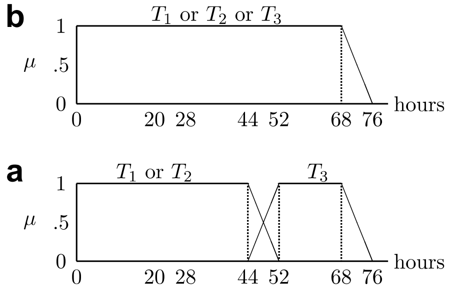

Fig. 5. Fuzzy temporal sequence hierarchy.

5. Temporal specification

Fuzzy hedges were introduced by Zadeh [21] to provide a linguistically understandable approach for modifying the membership values of fuzzy sets. Hedges are described by adverbs that intensify or weaken the property asserted by a fuzzy predicate. A fuzzy set *Tall* that describes height can be transformed into fuzzy sets that describe the notion of somewhat tall or very tall by

$$\begin{aligned} &Very\ Tall(x) = Tall(x)^2, \\ &Somewhat\ Tall(x) = \sqrt{Tall(x)}. \end{aligned}$$

These transformations increase or lessen intermediate membership values but do not affect the core of the original fuzzy set. Moreover, items with equal membership values in the original set have the same membership value in the transformed set. That is, if $Tall(x) = Tall(y)$, then $Very\ Tall(x) = Very\ Tall(y)$. Neither of these properties need hold for modifications to temporal intervals.

A temporal hedge focuses on a subinterval of the original constraint. For example, the hedges 'early in', 'in the middle of', and 'late in' designate subintervals near the beginning, near the middle, and near the end of a temporal interval. Fuzzy sets representing the relevancy constraints 'in the second day after' and 'late in the second day after' are shown in Fig. 6. Note that the core of the fuzzy set 'in the second day after' differs from the core of 'late in the second day after'. This shift matches the intuitive understanding of the meaning of 'late in the day'.

To compute the confidence of a rule in which the implicative relevancy constraint is modified by a temporal hedge requires the ability to determine the number of examples of the modified relationship. In [22,23] a binning process was used to summarize the distribution of examples for the optimization of the implication operator. A similar strategy will be employed to record the distribution examples based on the time between the satisfaction of the antecedent and that of the consequent. From this information we will be able to compute the frequency and confidence of both the original rules and the hedged variations.

A set of events that satisfies A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}, and C_k with the satisfaction of C_k occurring between 20 and 52 h after the satisfaction of the antecedent predicates is an example of the rule '$C_k : T_2$-after $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$'. Modifying the relevancy constraint with the hedge 'late in' changes the sets of examples as indicated in Table 1. Column 1 divides the support of the constraint T_2 into subintervals in which the membership function is increasing, constant, and decreasing. Columns 2 and 3 contrast the relationship between examples of T_2 and 'late in T_2' over these intervals.

To compute the confidence of a rule produced by temporal specification, we must record sufficient information to determine the number of examples and counterexamples of the new rule. The process will be illustrated using the rule

$$C_k : T_f\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle,$$

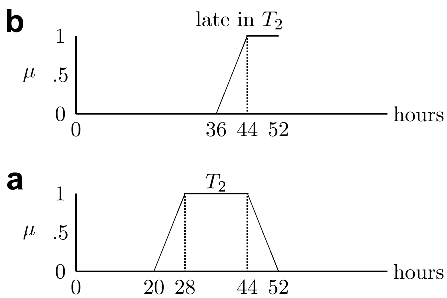

Fig. 6. Temporal hedge 'late'.

Table 1
Examples in T_2 and 'late in T_2'

Interval	T_2	Late in T_2
(20,28)	Partial example	Counterexample
[28,36)	Complete example	Counterexample
[36,44)	Complete example	Partial example
[44,52]	Partial example	Complete example

where T_f is defined by the trapezoidal fuzzy set

$$T_f(x) = \begin{cases} (x-\alpha_1)/(\alpha_2-\alpha_1) & \text{if } \alpha_1 \leqslant x < \alpha_2 \\ 1.0 & \text{if } \alpha_2 \leqslant x < \beta_2 \\ (x-\beta_1)/(\beta_1-\beta_2) & \text{if } \beta_2 \leqslant x < \beta_1 \\ 0 & \text{otherwise} \end{cases}$$

with support $[\alpha_1,\beta_1]$. We begin by subdividing the support of T_f into y subintervals. The value of y is chosen to provide a set of representative membership values and to define the bins for frequency counting. With y bins, the subintervals are

$$I_i = [\alpha_1 + (i-1)(\beta_1-\alpha_1)/y, \alpha_1 + i(\beta_1-\alpha_1)/y]$$

for $i = 1, \ldots, y-1$.

Let e_1, e_2, e_3, e_c be events that satisfy the rule

$$C_k : \text{late in} T_f\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle,$$

to some non-zero degree. Also let d_1 be the time between the satisfaction of the antecedent and the consequent and d_2 the time between the satisfaction of the predicates in the antecedent, as before. The degree of satisfaction of the predicates and the antecedent constraint $\alpha = A_{i_1,r_1}(e_i) \otimes A_{i_2,r_2}(e_j) \otimes A_{i_3,r_3}(e_k) \otimes C_k(e_c) \otimes W_g(d_2)$ is evaluated independently of the implicative constraint. The appropriate bin for this example is determined by finding the interval I_i with $d_1 \in I_i$ and is incremented by α.

Using frequency binning, the calculation of the number of examples is an approximation whose accuracy is determined by the number of bins employed. After the accumulation of the distribution of the examples, the number of examples of the original rule

$$C_k : T_f\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$$

is

$$\sum_{i=1}^{y} n_i \cdot T_f(\alpha_1 + (i+1/2)(\beta_1-\alpha_1)),$$

where n_i is the number of examples in the subinterval i. Using the 'late in T_f' membership function, the number of examples of the hedged rule

$$C_k : \text{late in} T_f\text{-after } \langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$$

is

$$\sum_{i=1}^{y} n_i \cdot (\text{late in} T_f(\alpha_1 + (i+1/2)(\beta_1-\alpha_1))).$$

The distribution of examples can be stored in the array Ex by adding another dimension with index ranging from 1 to y.

The distribution of counterexamples can be determined in a similar manner and stored in Cex. The number of examples and counterexamples is then used to compute the confidence of the modified rules.

Temporal specification is employed after the completion of the original learning process to adapt the rules to reflect the distribution of events within a relevancy constraint. After a rule '$C_k : T_2$- after $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$' has been accepted, the distribution of the frequencies for that rule is examined. If a preponderance of the frequencies occurs in the latter half of the table, the the validity of the rule 'C_k : late in T_2-after $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$' may be tested. Similarly, if the frequencies occur the initial entries in F, the rule 'C_k: early in T_2- after $\langle\{A_{i_1,r_1}, A_{i_2,r_2}, A_{i_3,r_3}\}, W_g\rangle$' may be tested.

Temporal specification permits the generation of rules with finer granularity while limiting the amount of additional work required. The original search may be carried out using a partition consisting of a small number of fuzzy sets based on temporal durations that naturally occur in the problem domain. Only rules that have been accepted during the hypothesis assessment phase of the learning process need be considered for refinement.

In the two refinement techniques presented in this paper, the family of constraints were defined by fuzzy partitions. This assumption simplified the determination of the examples and counterexamples in disjunctive generalization. The binning process described above can be used for both disjunctive generalization and temporal specification when the constraints are defined by an arbitrary family of fuzzy sets over the window of relevance because of the ability to determine the number of examples and counterexamples from the frequencies in the bin and the membership function of the derived constraint.

6. Impact of refinement

In this section we demonstrate the potential impact of rule refinement on both the flexibility of the rules that can be produced and their linguistic interpretability. We will use a simple traffic example to illustrate the effects of refinement. Attributes $\mathscr{A}_1, \ldots, \mathscr{A}_5$ indicate the traffic volume at five intersections and the consequent domain $\mathscr{C}$ represents the delay time at a ramp entering an autoroute. The linguistic terms describing the traffic conditions and temporal constraints, along with the partition of the implicative constraint, are given in Fig. 7.

A rule composed from these predicates and constraints predicts levels of congestion at the ramp based on the traffic conditions at three intersections. The interpretation of the rule

$$\textit{short delay} : \text{about } 10\,\text{min after } \langle\{\textit{medium}_1, \textit{medium}_2, \textit{high}_3\}, \text{within } 5\,\text{min}\rangle \tag{1}$$

is that if there is medium traffic volume at intersections 1 and 2, and high volume at intersection 3 within a 5-min period, then there will be a short delay at the ramp approximately 10 min later. There are 5,832 rules that can be constructed from the preceding predicates and constraints.

If two similar rules with adjacent implicative constraints are discovered during the rule learning phase, say rule (1) above and

	Linguistic Terms
Attribute $\mathcal{A}_i$:	$low_i, med_i, high_i$
Consequent $\mathcal{C}$:	*negligible*, *short*, *medium*, *long*
Antecedent Constraint:	'within 5 minutes', 'within 10 minutes'
Implicative Constraints:	'within 5 minutes after', 'about 10 minutes after',
	'about 30 minutes after'

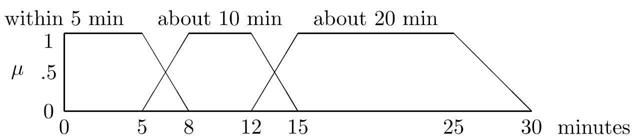

Fig. 7. Linguistic terms and implicative constraints.

Table 2
Linguistic descriptions of disjunctive constraints

Disjunction	Linguistic interpretation
'within 5 min after' or 'about 10 min after'	'within 10 min after'
'about 10 min after' or 'about 20 min after'	'between 10 and 20 min after'
'within 5 min after' or 'about 10 min after' or 'about 20 min after'	'within 20 min after'

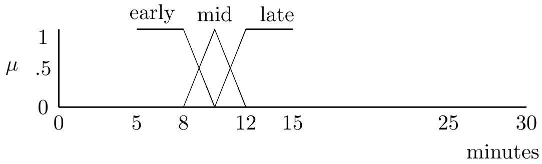

Fig. 8. Specification constraints from 'about 10 min after'.

$$\textit{short delay}: \text{about } 20\,\text{min after } \langle\{\textit{medium}_1, \textit{medium}_2, \textit{high}_3\}, \text{within } 5\,\text{min}\rangle,$$

these rules become candidates for disjunctive generalization. The support and confidence for the combined rule

$$\textit{short delay}: \text{about } 10\,\text{min or about } 20\,\text{min after } \langle\{\textit{medium}_1, \textit{medium}_2, \textit{high}_3\}, \text{within } 5\,\text{min}\rangle$$

is obtained as described in Section 4. The linguistic description of a rule produced by disjunctive generalization can be also modified to more clearly represent the derived constraint as shown in Table 2. The three derived constraints yield an additional 5832 possible rules that can be produced by the learning process.

Specification evaluates rules in which the examples are concentrated in subintervals of the implicative constraint. The possible specifications for the constraint 'within 10 min after' are shown in Fig. 8. The three specifications for each constraint add 17,496 possible rules that can be generated.

The refinement techniques quadruple the number of potential rules but require only information that is produced during the evaluation in the original learning phase. Restricting the refinement process to rules that exceed the support and confidence thresholds limits the computational resources needed to gain this increase in rule expressiveness.

7. Conclusion

Analyzing temporal information and searching for patterns utilize temporal constraints to indicate the relevancy of the occurrence of one event to another. Two techniques were presented to refine rules based on modifications to the temporal constraints. Disjunctive generalization produces more general rules by merging adjacent constraints in the partition of the window of relevance. Temporal specification uses linguistic hedges to reduce the duration of the implicative constraint while maintaining the interpretability of the rule. Both of these techniques increase the expressibility of the rules without the computational cost associated with performing the search for examples with a large family of constraints. The information needed for the refinements can be obtained during process of learning rules with the original partition of the constraints and refinements need be considered only on the subset of rules accepted by the original learning algorithm.

References

[1] K.P. Adlassnig, C. Combi, A.K. Das, E.T. Keravnou, G. Pozzi, Temporal representation and reasoning in medicine: research directions and challenges, Artificial Intelligence in Medicine 38 (2006) 101–113.
[2] J.H. Lee, H.L. Kwang, An extension of association rules using fuzzy sets, in: Proceedings of IFSA 1997, Prague, 1997, pp. 399–402.
[3] K.C.C. Chan, W.-H. Au, An effective algorithm for mining interesting quantitative association rules, in: Selected Areas in Cryptography, 1997, pp. 88–90.
[4] M. Delgado, N. Marín, D. Sánchez, M.A. Vila, Fuzzy association rules: General model and applications, IEEE Transactions on Fuzzy Systems (2003) 214–225.

[5] E. Hullermeier, Y. Yi, In defense of fuzzy association rules, IEEE Transactions on Systems Man and Cybernetics: Part B 37 (2) (2007) 1039–1043.
[6] R. Agrawal, R. Srikant, Mining sequential patterns, in: P.S. Yu, A.S.P. Chen (Eds.), Eleventh International Conference on Data Engineering, IEEE Computer Society Press, Taipei, Taiwan, 1995, pp. 3–14.
[7] R. Srikant, R. Agrawal, Mining sequential patterns: generalizations and performance improvements, in: P.M.G. Apers, M. Bouzeghoub, G. Gardarin (Eds.), Proceedings of the 5th International Conference on Extending Database Technology, EDBT, vol. 1057, Springer-Verlag, 1996, pp. 3–17.
[8] H. Mannila, H. Toivonen, A.I. Verkamo, Discovering frequent episodes in sequences, in: Proceedings of the International Conference on Knowledge Discovery and Data Mining, AAAI Press, Montreal, 1995, pp. 210–215.
[9] B. Ozden, S. Ramaswamy, A. Silberschatz, Cyclic association rules, in: ICDE, 1998, pp. 412–421.
[10] J.M. Ale, G.J. Rossi, An approach to discovering temporal association rules, in: Proceedings of the 2000 ACM Symposium on Applied Computing, Como, Italy, 2000, pp. 294–300.
[11] J. Han, W. Gong, Y. Yin, Mining segment-wise periodic patterns in time-related databases, in: Proceedings of the International Conference on Knowledge and Data Mining, New York, 1998, pp. 1–12.
[12] R.B.V. Subramanyam, A. Goswami, A fuzzy data mining algorithm for incremental mining of quantitative sequential patterns, International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems 13 (6) (2005) 633–652.
[13] J.-Y. Jiang, W.-J. Lee, S.-J. Lee, Mining calendar-based asynchronous perdiocal association rules with fuzzy calendar constraints, in: Proceedings of the 2005 IEEE International Conference on Fuzzy Systems, Reno, NV, 2005, pp. 773–775.
[14] T.A. Sudkamp, R. Hammell, Discovering imprecise temporal associations, in: Proceedings of the NAFIPS-2003 Conference, Chicago, 2003, pp. 99–104.
[15] T. Sudkamp, Discovery of fuzzy temporal associations in multiple data streams, in: F. Hoffman, M. Koppen, F. Klawonn, R. Roy (Eds.), Advances in Soft Computing, LNCS, Springer-Verlag, Berlin, 2005, pp. 1–13.
[16] G. Vincenti, R.J.H. II, G. Trajkovski, Scouting for imprecise temporal associations to support effectiveness for drugs during clinical trials, in: Proceedings of the NAFIPS-05, 2005, pp. 171–176.
[17] T. Sudkamp, Examples, counterexamples, and measuring fuzzy associations, Fuzzy Sets and Systems 146 (2) (2005) 57–71.
[18] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, in: P. Buneman, S. Jajodia (Eds.), Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, 1993, pp. 207–216.
[19] R. Agrawal, R. Srikant, Fast algorithms for mining association rules, in: J.B. Bocca, M. Jarke, C. Zaniolo (Eds.), Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, Santiago, Chile, 1994, pp. 487–499.
[20] D. Dubois, E. Hullermeier, H. Prade, A note on quality measures for fuzzy association rules, in: T. Bilgiç, B.D. Baets, O. Kaynak (Eds.), Proceedings of the Eleventh IFSA World Congress, LNAI 2715, Istanbul, 2003, pp. 346–353.
[21] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning: Part 1, Information Sciences 8 (1975) 199–249.
[22] D. Dubois, H. Prade, T. Sudkamp, On the representation, measurement, and discovery of fuzzy associations, IEEE Transactions on Fuzzy Systems 13 (2) (2005) 250–262.
[23] M. Serrurier, D. Dubois, H. Prade, T. Sudkamp, G. Richard, Learning first order rules with their implication operator, in: IPMU–2004, Perugia, Italy, 2004, pp. 331–838.